Analytical symmetry detection in protein assemblies. I. Cyclic symmetries

Guillaume Pagès, Elvira Kinzina, Sergei Grudinin

Analytical symmetry detection in protein assemblies. I. Cyclic symmetries

Guillaume Pagès[a], Elvira Kinzina,[b], Sergei Grudinin[a]

[a]*Inria, Univ. Grenoble Alpes, CNRS, Grenoble INP, LJK, Grenoble, 38000, France*
[b]*Moscow Institute of Physics and Technology, Dolgoprudny, 141701, Russia*

Abstract

Symmetry in protein, and, more generally, in macromolecular assemblies is a key point to understand their structure, stability and function. Many symmetrical assemblies are currently present in the Protein Data Bank (PDB) and some of them are among the largest solved structures, thus an efficient computational method is needed for the exhaustive analysis of these. The cyclic symmetry groups represent the most common assemblies in the PDB. These are also the building blocks for higher-order symmetries. This paper presents a mathematical formulation to find the position and the orientation of the symmetry axis in a cyclic symmetrical protein assembly, and also to assess the quality of this symmetry. Our method can also detect symmetries in partial assemblies.

We provide an efficient C++ implementation of the method and demonstrate its efficiency on several examples including partial assemblies and pseudo symmetries. We also compare the method with two other published techniques and show that it is significantly faster on all the tested examples. Our method produces results with a machine precision, its cost function is solely based on 3D Euclidean geometry, and most of the operations are performed analytically. The method is available at http://team.inria.fr/nano-d/software/ananas. The graphical user interface of the method built for the SAMSON platform is available at http://samson-connect.net.

Keywords: Point-Group Symmetry, Protein Structure, Protein Assemblies, Continuous Optimization

Email address: sergei.grudinin@inria.fr (Sergei Grudinin)

1. Introduction

Symmetrical protein complexes are very common in nature, as can be seen from many symmetrical structures deposited to the Protein Data Bank (PDB) [1]. Indeed, it appears that symmetrical assemblies have many advantages compared to individual proteins [2, 3] and thus many of these have been selected during evolution. Thus, there is a considerable interest in studying the structures and mechanisms of formation of symmetrical assemblies [4, 5, 6, 7, 8, 9]. In particular, it has been demonstrated that molecular symmetries are important for evolution [3, 10], stability [11], folding and function [12].

The growing amount of data from constantly solved structures of macromolecules together with even bigger amount of data obtained with molecular dynamics simulations require fast and robust computational tools for the processing of these data. For example, some tools have been developed to detect and assess internal cyclic symmetries, based either on protein sequence [13], structure [14, 15], or both [16]. All these have a common idea of comparing a protein structure with a rotated version of itself. Another set of methods for the continuous chirality and symmetry analysis has been developed by David Avnir and colleagues [17, 18, 19] and also by Michel Petitjean [20], however these do not seem computationally suitable for processing large amounts of macromolecular data, specifically those from PDB. On the other hand, determining a symmetry group of a molecular assembly, finding its axes of symmetry, and assessing the quality of this symmetry are the essential steps in analysis of structural molecular data. For example, a basic analysis method has been proposed by Emmanuel Levy [2], but this is not fully satisfying due to its limited precision imposed by a set of discretely chosen axes with about 6 degrees of angular step, which results in total of about 600 axes. Also, this method is significantly more time consuming compared to the one presented below.

Inspired by the quaternion arithmetic applied to the best superposition of a set of points [21, 22, 23] together with our recent developments [24, 25], we propose a new symmetry measure and an analytical method to find the best symmetry axes of a symmetrical assembly possessing a cyclic C_n symmetry. We should specifically emphasize that our method assumes that the point-to-point correspondence between the different subunits of the assembly is

known. Many algorithms are available to establish this correspondence, e.g. using sequence or structure alignment between different subunits, and this is not the problem that we address here. Our method produces results with a machine precision, its cost function is solely based on 3D Euclidean geometry, and most of the operations are performed analytically. This makes our method extremely fast and particularly suitable for exhaustive analysis of PDB data.

Below we explain how to compute this symmetry measure and the axis of rotation of a cyclic C_n assembly. More precisely, we find the axis of rotation by solving a constrained quadratic optimization problem. As unknowns of the optimization problem, we include both the position and the direction of the symmetry axis. This allows, for example, to reconstruct the complete multi-subunit C_n assembly and its symmetry axis having just two subunits, or, to compute the symmetry measure for macromolecules with non-crystallographic point groups.

2. Methods

2.1. Notations

Throughout the paper we will be generally dealing with 3×3 matrices and 3-vectors. Therefore, for linear algebra operations we will stick to the following notation. Bold upper case letters (i.e. $\mathbf{A}$) will denote matrices, bold lower case letters (i.e. $\mathbf{b}$) will denote vectors, and normal weight lower case letters (i.e. c) will denote scalars. For trigonometric operations and illustrations we will also use an arrow notation for 3-vectors, such as $\vec{v}$. A rotation by an angle α about an axis $\vec{v}$ will be noted $R(\alpha, \vec{v})$.

2.2. Quaternion arithmetic

It is very convenient to express three-dimensional rotations using quaternion arithmetic. Thus, we will give a brief summary of it here. More informations on quaternions can be found in our previous paper [24], for example. We consider a quaternion Q as a combination of a scalar s with a 3-component vector $\mathbf{q} = \{q_x, q_y, q_z\}^T$, $Q = [s, \mathbf{q}]$. Quaternion algebra defines multiplication, division, inversion and norm, among other operations. The product of two quaternions $Q_1 = [s_1, \mathbf{q}_1]$ and $Q_2 = [s_2, \mathbf{q}_2]$ is a quaternion and can be expressed through a combination of scalar and vector products,

$$\begin{aligned} Q_1 \cdot Q_2 &= [s_1, \mathbf{q}_1] \cdot [s_2, \mathbf{q}_2] \\ &= [s_1 s_2 - (\mathbf{q}_1 \cdot \mathbf{q}_2), s_1 \mathbf{q}_2 + s_2 \mathbf{q}_1 + (\mathbf{q}_1 \times \mathbf{q}_2)] \,. \end{aligned} \tag{1}$$

The squared norm of a quaternion Q is given as $|Q|^2 = s^2 + \mathbf{q} \cdot \mathbf{q}$, and a unit quaternion is a quaternion with its norm equal to 1. Finally, a unit quaternion $\hat{Q}$ corresponding to a rotation by an angle α around a unit axis $\mathbf{v}$ is given as $\hat{Q} = [\cos\frac{\alpha}{2}, \mathbf{v}\sin\frac{\alpha}{2}]$, and its inverse is $\hat{Q}^{-1} = [\cos\frac{\alpha}{2}, -\mathbf{v}\sin\frac{\alpha}{2}]$.

2.3. Root mean square deviation

The root mean square deviation (RMSD) is one of the most widely used similarity criteria in structural biology and bioinformatics. We will stick to this measure throughout the manuscript, as it is very powerful, easy to understand and also because it can be computed very efficiently. For our particular needs we will use the definition of RMSD between two ordered sets of points, where each point has an equal contribution to the overall RMSD loss. More precisely, given a set of N points $A = \{\mathbf{a}_i\}_N$ and $B = \{\mathbf{b}_i\}_N$, the RMSD between them is defined as

$$\mathrm{RMSD}(A, B)^2 = \frac{1}{N} \sum_{1 \leq i \leq N} |\mathbf{a}_i - \mathbf{b}_i|^2 . \tag{2}$$

2.4. The RMSD master equation

Let us formally define the problem of the best superposition of two rigid molecules. Suppose that the operator associated with a rotation about axis $\vec{v}$ by an angle α may be labelled $\hat{R}(\alpha, \vec{v})$. Let us also suppose that the operator associated with a translation by a vector $\vec{u}$ is labelled $\hat{T}(\vec{u})$. We should mention that we have borrowed the presented formalism from the molecular docking methods [4], where it appears very useful.

Let $\mathbf{u}$ be a translation vector and $\hat{Q} \equiv [s, \mathbf{q}]$ a rotation quaternion corresponding to the operators $\hat{T}(\vec{u})$, and $\hat{R}(\alpha, \vec{v})$, respectively. We apply these to an assembly A composed of N_s subunits with N_a atoms at positions $A = \{\mathbf{a}_{i,j}\}_{N_s,N_a}$ with $\mathbf{a}_{i,j} = \{x_{i,j}, y_{i,j}, z_{i,j}\}^T$, and compare the result with the positions of a molecule B with the same number of subunits and atoms at positions $B = \{\mathbf{b}_{i,j}\}_{N_s,N_a}$ with $\mathbf{b}_{i,j} = \{x'_{i,j}, y'_{i,j}, z'_{i,j}\}^T$. Using a similar reasoning to what we presented in our previous work [24], the RMSD between new positions of A and B in the reference frame bound to the center of mass (COM) of A is given as

$$\mathrm{RMSD}^2(\hat{T}(\vec{u})\hat{R}(\alpha, \vec{v})A, B) = \frac{4}{N}\mathbf{q}^T\mathbf{I}'\mathbf{q} + 4s\mathbf{q}^T\mathbf{x}_\perp + \mathbf{u}^2 + 2\mathbf{u}^T\mathbf{x}_m + x_s. \tag{3}$$

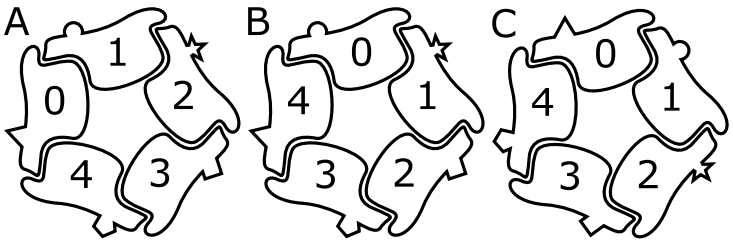

Figure 1: A - An assembly with an almost perfect C_5 symmetry. Each subunit is represented with an outline whose shapes are slightly different from each other. B - The 1-permuted version of this assembly, the shapes are the same as in A but the labelling is different. C - The rotated version of the assembly A by an angle $2\pi/5$.

Here, the modified inertia tensor $\mathbf{I}'$ is given as

$$\mathbf{I}' = \begin{pmatrix} \sum(y_{i,j}y'_{i,j} + z_{i,j}z'_{i,j}) & -\sum(x'_{i,j}y_{i,j} + x_{i,j}y'_{i,j})/2 & -\sum(x'_{i,j}z_{i,j} + x_{i,j}z'_{i,j})/2 \\ -\sum(x_{i,j}y'_{i,j} + x'_{i,j}y_{i,j})/2 & \sum(x_{i,j}x'_{i,j} + z_{i,j}z'_{i,j}) & -\sum(y'_{i,j}z_{i,j} + y_{i,j}z'_{i,j})/2 \\ -\sum(x_{i,j}z'_{i,j} + x'_{i,j}z_{i,j})/2 & -\sum(y_{i,j}z'_{i,j} + y'_{i,j}z_{i,j})/2 & \sum(x_{i,j}x'_{i,j} + y_{i,j}y'_{i,j}) \end{pmatrix}. \tag{4}$$

The vectors $\mathbf{x}_\perp$, $\mathbf{x}_m$, and the scalar x_s are

$$\begin{aligned} \mathbf{x}_\perp &= \sum_{i,j} \mathbf{b}_{i,j} \times \mathbf{a}_{i,j}/N \\ \mathbf{x}_m &= -\sum_{i,j} \mathbf{b}_{i,j}/N \\ x_s &= \sum_{i,j} (\mathbf{a}_{i,j} - \mathbf{b}_{i,j})^2/N. \end{aligned} \tag{5}$$

Below, we will analytically determine axes that correspond to the chosen C_n symmetries by minimizing eq. 3 with proper constraints.

We should specifically mention that if the coordinates of A and B are only different by a permutation of their indexes, as it happens in many practical cases of symmetry detection described below, then the vector $\mathbf{x}_m$ becomes zero. This uncouples the RMSD master equation with respect to the translation and rotation and greatly simplifies many corresponding equations. More precisely, minimization of RMSD with respect to $\mathbf{u}$ in this case gives a trivial solution $\mathbf{u} = 0$.

2.5. Working with molecular assemblies

As we work with assemblies composed of macromolecules such as proteins, it is convenient to introduce an intermediate level of structural hierarchy

between the *complete* assembly and its N atoms. Let us consider a molecular assembly as a list of N_s subunits, each containing N_a atoms such that $N = N_a N_s$. The RMSD between two assemblies is then

$$\mathrm{RMSD}(A,B)^2 = \frac{1}{N} \sum_{0 \le i < N_s} \sum_{0 \le j < N_a} |\mathbf{a}_{i,j} - \mathbf{b}_{i,j}|^2 \,. \tag{6}$$

We can assume that every subunit has the same number of *reference* points. Technically, we achieve it by performing a multiple sequence alignment of the subunits and keeping only the aligned parts for the subsequent analysis. More precisely, the reference points are located at the positions of the aligned C_α atoms. This makes our method robust against various inconsistencies in the input data.

It will be convenient to assume that the subunits in the assembly are labelled with integers modulo of n, i.e. i and $i+n$ refer to the same subunit. Let us also assume that the labelling is *sequential*, meaning that the subunit i is located *between* the subunits $i-1$ and $i+1$. Finally, let us define a *k-permuted* version A^k of the assembly A by

$$\mathbf{a}^k_{i,j} = \mathbf{a}_{i+k,j} \tag{7}$$

Note that according to this definition, A is equal to its *0-permuted* version, and a *k-permuted* assembly matches itself rotated by $2k\pi/n$. If the subunits are not labelled sequentially, finding the permutation between the subunits, that is associated with every rotation operator, is not straightforward. Our initial approach consisted in projecting the centers of mass of the different subunits on the plane orthogonal to the principal eigenvector of the inertia matrix of the assembly, and then reordering the subunits according to this projection. During the second part of this work [26], we developed a much more general and robust method that automatically determines the permutations between the subunits for each rotation operator in a certain symmetry group including cyclic, dihedral and cubic cases.

2.6. Complete C_n assembly

Let us first assume that we have as input a *complete* cyclic assembly, for which we want to assess the quality of the cyclic symmetry. A cyclic symmetry group of order n can be uniquely described with its symmetry axis $\vec{v}$, the position of this axis, and its order n. As it is explained above, the translational part of the RMSD master equation 3 in this case is equal to zero,

because the two sets of points are permutations of each other. The angles of the rotation operators are constrained to be $\{k\omega\}_{0\leq k<n}$ with $\omega = 2\pi/n$. To determine the *quality* of a rotation axis $\vec{v}$, we compute the RMSD between the assembly rotated by an angle of $k\omega$ (see Fig. 1C) and a *k-permuted* version of the original assembly (see Fig. 1B), as it is shown in Figure 1. This RMSD will thus be our *symmetry measure*.

The quaternion representation of the k^{th} C_n symmetry operator is given as

$$\hat{Q^k} \equiv [s, \mathbf{q}] = [\cos\frac{k\omega}{2}, \sin\frac{k\omega}{2}\mathbf{v}], \tag{8}$$

with $0 \leq k < n$. According to the RMSD master equation, with $B = A^k$ and $\mathbf{u} = 0$, we obtain

$$\mathrm{RMSD}^2(\hat{R}(k\omega, \vec{v})A, A^k) = \frac{4}{N}\mathbf{q}^T\mathbf{I}'_k\mathbf{q} + 4s\mathbf{q}^T\mathbf{x}_{k\perp} + x_{ks}. \tag{9}$$

Here

$$\mathbf{I}'_k = \begin{pmatrix} \sum(y_{i,j}y_{k+i,j} + z_{i,j}z_{k+i,j}) & -\sum(x_{k+i,j}y_{i,j} + x_{i,j}y_{k+i,j})/2 & -\sum(x_{k+i,j}z_{i,j} + x_{i,j}z_{k+i,j})/2 \\ -\sum(x_{i,j}y_{k+i,j} + x_{k+i,j}y_{i,j})/2 & \sum(x_{i,j}x_{k+i,j} + z_{i,j}z_{k+i,j}) & -\sum(y_{k+i,j}z_{i,j} + y_{i,j}z_{k+i,j})/2 \\ -\sum(x_{i,j}z_{k+i,j} + x_{k+i,j}z_{i,j})/2 & -\sum(y_{i,j}z_{k+i,j} + y_{k+i,j}z_{i,j})/2 & \sum(x_{i,j}x_{k+i,j} + y_{i,j}y_{k+i,j}) \end{pmatrix}, \tag{10}$$

and

$$\begin{aligned} \mathbf{x}_{k\perp} &= \sum_{i,j} \mathbf{a}_{k+i,j} \times \mathbf{a}_{i,j}/N \\ x_{ks} &= \sum_{i,j} (\mathbf{a}_{i,j} - \mathbf{a}_{k+i,j})^2/N. \end{aligned} \tag{11}$$

Finding the best rotation axis reduces to the following optimization problem,

$$\begin{aligned} \min_{\mathbf{v}} \quad & \mathrm{RMSD}^2(\mathbf{v}) = \mathbf{v}^T\mathbf{A}_k\mathbf{v} + \mathbf{d}_k^T\mathbf{v} + f_k \\ & \text{s.t.} \qquad \mathbf{v}^T\mathbf{v} = 1, \end{aligned} \tag{12}$$

where

$$\begin{aligned} \mathbf{A}_k &= \frac{4}{N}\sin^2\frac{k\omega}{2}\mathbf{I}'_k \\ \mathbf{d}_k &= 2\sin(k\omega)\mathbf{x}_{k\perp} \\ f_k &= x_{ks}. \end{aligned} \tag{13}$$

Equations 12-13 formulate a minimization problem to find an axis corresponding to a particular rotation operator with a fixed rotation angle. However, our goal is to determine the axis that is the best for all the rotation operators. We can thus sum up the above expressions for every k, as the axis $\vec{v}$, which we are seeking for, is *the same* for all the rotation operators. Finally, finding the best axis of symmetry for a C_n group is equivalent to solving the following *trust-region subproblem*,

$$\begin{aligned} \min_{\mathbf{v}} \quad & \mathbf{v}^T \sum_{k=1}^{k<n} \mathbf{A}_k \mathbf{v} + \sum_{k=1}^{k<n} \mathbf{d}_k^T \mathbf{v} + \sum_{k=1}^{k<n} f_k \\ \text{s.t.} \quad & \mathbf{v}^T \mathbf{v} = 1. \end{aligned} \tag{14}$$

This is a well-studied optimization problem. It can be efficiently solved with a number of different methods. In our case, the dimensionality of the problem is very low and thus we have chosen the solver based on the Sorensen method [27], which typically converges to machine precision in 3 - 10 iterations in our case. Equation 14 constitutes the first principal result of this work. We should note that in a particular C_2 case, the $\mathbf{d}_k^T$ coefficients vanish and the solution of the problem 14 reduces to the smallest eigenvector of matrix $\sum_{k=1}^{k<n} \mathbf{A}_k$.

2.7. C_n assembly with missing subunits

Some examples of molecular assemblies with presumably cyclic symmetry are not complete and have missing subunits. This automatically raises two questions: what should be the order of the complete assembly and how to reconstruct it? The ability to find the rotation operator that produces the smallest RMSD between the present subunits with a constrained angle answers these two questions. To determine the best order of the cyclic symmetry, we can simply exhaustively test all the different possible orders by changing the constraint on the angle of the rotation operator, as it is given by equation 8, and then solving the RMSD master equation 3. Once this step is done, we obtain the order and the axis of symmetry, which makes the reconstruction of the complete assembly trivial. However, in this case, we need to solve the full version of the RMSD master equation, since the translational component of RMSD is not null.

To determine the axis of the rotation operator, similarly to the case with the complete assembly considered above, we will compare the rotated version of the *partial* assembly with its permuted version. We should mention that

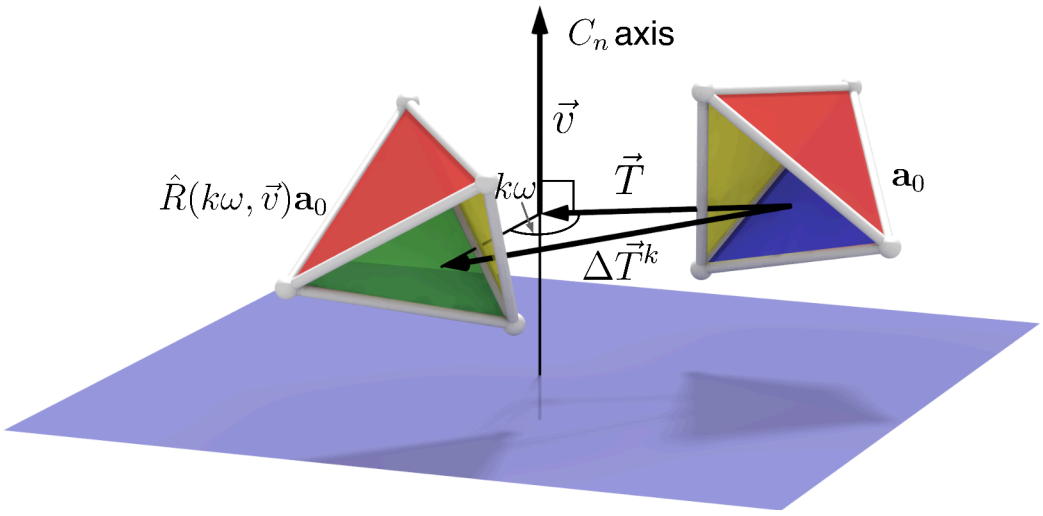

Figure 2: Illustration of the rotation of a subunit $\mathbf{a}_0$. The original and rotated versions of $\mathbf{a}_0$ are represented as tetrahedrons having four differently colored faces (red, green, blue and yellow). For the clarity of the representation, the green face was removed from $\mathbf{a}_0$ and the blue face was removed from $\hat{R}(k\omega, \vec{v})\mathbf{a}_0$. The $\vec{T}$ vector connects the COM of $\mathbf{a}_0$ with the symmetry axis. The $\Delta\vec{T}$ vector connects the COM of $\mathbf{a}_0$ with the COM of $\hat{R}(k\omega, \vec{v})\mathbf{a}_0$.

in the case of partial assembly we assume the sequential order of the input subunits. If it is not the case, the order has to be specified manually, since the performance of the automatic procedure for the order perception is largely affected by the missing subunits. Let us assume that the subunit $\mathbf{a}_0 = \{x_{0,j}, y_{0,j}, z_{0,j}\}^T_{(1\leq j\leq N_a)}$ is present. Let us label the vector that connects the COM of the $\mathbf{a}_0$ subunit with the symmetry axis $\vec{v}$, and which is perpendicular to it, as $\vec{T}$. Following Figure 2, the translation vector $\Delta\vec{T^k}$ that connects the COM of $\mathbf{a}_0$ with the COM of $\hat{R}(k\omega, \vec{v})\mathbf{a}_0$ is

$$\Delta\mathbf{T}^k = (1 - \cos(k\omega))\,\mathbf{T} - \sin(k\omega)\mathbf{v} \times \mathbf{T}. \tag{15}$$

The squared 2-norm of this vector is given as

$$\left(\Delta\mathbf{T}^k\right)^2 = 4\sin^2\frac{k\omega}{2}\mathbf{T}^2. \tag{16}$$

Now we are ready to substitute the rotation quaternion from equation 8 and the obtained translation vector into the RMSD master equations 3. The RMSD is now a function of $\mathbf{T}$ and $\mathbf{v}$ vectors. Keeping the quaternion representation from equation 8, we obtain

$$\begin{aligned}\mathrm{RMSD}_k^2 =& \frac{4}{N}\mathbf{q}^T\mathbf{I}'\mathbf{q} + 4s\mathbf{q}^T\mathbf{x}_\perp + 4\sin^2\frac{k\omega}{2}\mathbf{T}^2 \\ &+ \left((1 - \cos(k\omega))\,\mathbf{T} - \sin(k\omega)\mathbf{v} \times \mathbf{T}\right)^T\mathbf{x}_m + x_s,\end{aligned} \tag{17}$$

which reduces to the following optimization problem,

$$\begin{aligned}\min_{\mathbf{v},\mathbf{T}} \quad & \mathrm{RMSD}_k^2(\mathbf{v},\mathbf{T}) = \mathbf{v}^T\mathbf{A}_k\mathbf{v} + b_k\mathbf{T}^2 + \mathbf{v}^T\mathbf{C}_k\mathbf{T} \\ & \qquad\qquad + \mathbf{d}_k^T\mathbf{v} + \mathbf{e}_k^T\mathbf{T} + f_k \\ \text{s.t.} \quad & \begin{cases}\mathbf{v}^T\mathbf{v} = 1 \\ \mathbf{v}^T\mathbf{T} = 0\end{cases}.\end{aligned} \tag{18}$$

Here, the coefficients are given as

$$
\begin{aligned}
\mathbf{A}_k &= \frac{4}{N}\sin^2(\frac{k\omega}{2})\mathbf{I}' \\
b_k &= 4\sin^2(\frac{k\omega}{2}) \\
\mathbf{C}_k &= 2\sin(k\omega)\begin{pmatrix} 0 & \mathbf{x}_{m3} & -\mathbf{x}_{m2} \\ -\mathbf{x}_{m3} & 0 & \mathbf{x}_{m1} \\ \mathbf{x}_{m2} & -\mathbf{x}_{m1} & 0 \end{pmatrix} \\
\mathbf{d}_k &= 2\sin(k\omega)\mathbf{x}_\perp \\
\mathbf{e}_k &= -4\sin^2(\frac{k\omega}{2})\mathbf{x}_m \\
f_k &= x_s,
\end{aligned}
\tag{19}
$$

which follows from the substitution of eqs. (8) and (15) into the RMSD master equation 3. In the above equation the definitions of matrix $\mathbf{I}'$, and vectors $\mathbf{x}_\perp$ and $\mathbf{x}_m$ are taken from equations 4 and 5 with the substitutions of $\mathbf{a} = \mathbf{a}_0$ and $\mathbf{b} = \mathbf{a}_\mathbf{k}$. At this point, vectors $\mathbf{v}$ and $\mathbf{T}$ are defined independently from the index k, thus we can sum up equation 18 for all k corresponding to the present subunits, and provide the global coefficients that will define the overall symmetry measure $\mathrm{RMSD}^2(\mathbf{v},\mathbf{T}) = \sum_k \mathrm{RMSD}_k^2(\mathbf{v},\mathbf{T})$ as

$$
\begin{aligned}
\mathbf{A} &= \sum_k \mathbf{A}_k \\
b &= \sum_k b_k \\
\mathbf{C} &= \sum_k \mathbf{C}_k \\
\mathbf{d} &= \sum_k \mathbf{d}_k \\
\mathbf{e} &= \sum_k \mathbf{e}_k \\
f &= \sum_k x.
\end{aligned}
\tag{20}
$$

Using the *Lagrangian formalism*, we can introduce two Lagrange multipliers λ_1 and λ_2 with the Lagrangian function $L(\mathbf{v},\mathbf{T},\lambda_1,\lambda_2)$ that incorporates two

equality constraints from eq. (18) as

$$\begin{aligned} L(\mathbf{v},\mathbf{T},\lambda_1,\lambda_2) =& \mathbf{v}^T\mathbf{A}\mathbf{v} + b\mathbf{T}^T\mathbf{T} + \mathbf{v}^T\mathbf{C}\mathbf{T} \\ &+ \mathbf{d}^T\mathbf{v} + \mathbf{e}^T\mathbf{T} + f + \lambda_1(\mathbf{v}^T\mathbf{v}-1) + \lambda_2\mathbf{v}^T\mathbf{T}. \end{aligned} \tag{21}$$

Here, matrix $\mathbf{A}$ is symmetric and positive definite, while matrix $\mathbf{C}$ is skew-symmetric. Setting the gradient $L_{\mathbf{T}}$ to zero gives

$$\begin{aligned} &(\mathbf{C}^T + \lambda_2\mathbf{E}_3)\mathbf{v} + \mathbf{e} + 2b\mathbf{T} = 0 \\ &\text{s.t.} \quad \begin{cases} \mathbf{v}^T\mathbf{v} = 1 \\ \mathbf{v}^T\mathbf{T} = 0 \end{cases}, \end{aligned} \tag{22}$$

where $\mathbf{E}_3$ is a 3×3 identity matrix. Left-multiplying the first equation by $\mathbf{v}^T$, we obtain

$$\lambda_2 + \mathbf{e}^T\mathbf{v} = 0. \tag{23}$$

Therefore, we can determine the first unknown vector $\mathbf{T}$ as

$$\mathbf{T} = -\frac{1}{2b}\left(\mathbf{e} + \mathbf{C}^T\mathbf{v} - (\mathbf{e}^T\mathbf{v})\mathbf{v}\right). \tag{24}$$

Now, substituting it to the minimization function $\mathrm{RMSD}^2(\mathbf{v},\mathbf{T})$, we obtain

$$\begin{aligned} \mathrm{RMSD}^2(\mathbf{v},\mathbf{T}) =& \mathbf{v}^T\mathbf{A}\mathbf{v} + \mathbf{d}^T\mathbf{v} + f \\ &+ \frac{1}{4b}(-\mathbf{e}^2 + 2\mathbf{e}^T\mathbf{C}\mathbf{v} - \mathbf{v}^T\mathbf{C}\mathbf{C}^T\mathbf{v} + \mathbf{v}^T\mathbf{e}\mathbf{e}^T\mathbf{v}). \end{aligned} \tag{25}$$

As a result, our initial optimization problem 18 reduces to the following form,

$$\begin{aligned} &\min_{\mathbf{v}} \mathbf{v}^T\mathbf{X}\mathbf{v} + \mathbf{y}^T\mathbf{v} + z \\ &\text{s.t.} \quad \mathbf{v}^T\mathbf{v} = 1 \end{aligned}, \tag{26}$$

where the coefficients $\mathbf{X}$, $\mathbf{y}$, and z are given as

$$\begin{aligned} \mathbf{X} &= \mathbf{A} + \frac{1}{4b}(-\mathbf{C}\mathbf{C}^T + \mathbf{e}\mathbf{e}^T) \\ \mathbf{y} &= \frac{1}{2b}\mathbf{C}^T\mathbf{e} \ + \mathbf{d} \\ z &= -\frac{1}{4b}\mathbf{e}^T\mathbf{e} + f. \end{aligned} \tag{27}$$

This is once again the previously introduced *trust-region subproblem*. Equations 26-27 constitute the second principal result of this work.

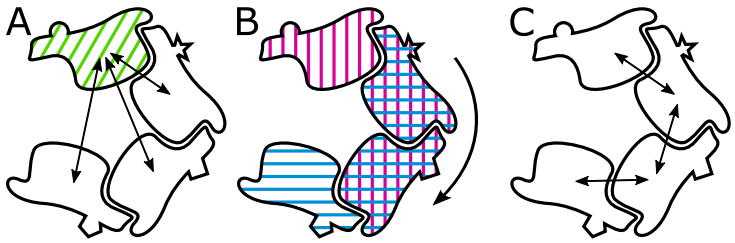

Figure 3: Assembly with C_5 symmetry and a missing subunit. A - The arrows show the comparisons made using the subunit with diagonal lines as the master subunit $\mathbf{a}_0$. B - With the same assembly, one rotation operator has been chosen, the part with vertical lines represents the *virtual reference* subunit and the part with horizontal lines is the *virtual target* subunit (they overlap). C - The arrows shows the comparisons resulting from the subunits' definition made in B.

2.8. Choice of Symmetry Measure

While the symmetry measure for the complete cyclic assembly is trivial and unique, there are multiple choices of this for partial assemblies. Indeed, in the later case the determined symmetry axis depends on the choice of the *master* subunit $\mathbf{a}_0$ and also on the performed comparisons. Figure 3A shows the simplest choice of the symmetry measure, where the *master* subunit is progressively superposed with every other subunit, while the other ones are only superposed with $\mathbf{a}_0$. The symmetry measure then reports the mean RMSD corresponding to the symmetry-constrained superposition of the *master* subunit with the rest of the assembly. Ideally, we would like to compare every subunit to every other subunit. However, this type of comparison makes the RMSD master equation 3 intractable using the presented techniques.

Therefore, orthogonally to the first approach, we can also choose a symmetry rotation operator and compare all the subunits that are superposed by this operator, as it is shown in Figures 3B-C. This can be seen as a redefinition of subunits by grouping all the matching subunits into new larger *virtual* subunits. More precisely, we can introduce a *virtual reference* subunit composed of all subunits that will be matched with other subunits by this operator. We can also introduce a *virtual target* subunit composed of all the subunits to which the *virtual reference* subunit matches. These *virtual* subunits are automatically perceived to contain the maximum number of individual subunits. We then compare these two *virtual* subunits, as it is shown in Figure 3B. This way, we uniquely define the symmetry measure for

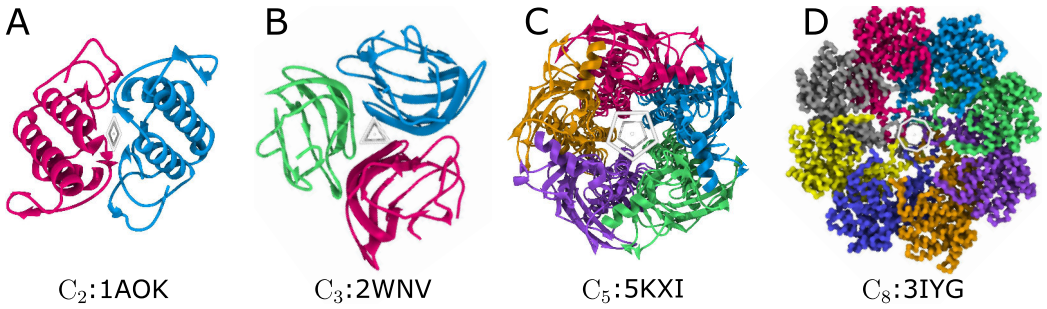

Figure 4: Example of symmetry detection of four pseudo-symmetrical assemblies with C_2, C_3, C_5, and C_8 symmetries. The determined symmetry axes are orthogonal to the screen. The order n of each axis is represented with a regular n-gone, except of order 2 shown with a rhombus. The corresponding RMSD symmetry measures are 1.406 Å, 2.226 Å, 1.613 Å, and 2.736 Å, respectively. This illustration and all the illustrations below were produced in SAMSON (www.samson-connect.net).

one rotation operator. This will report the mean RMSD corresponding to the subunits superposed by this operator.

The released version of our method implements the rotation operator approach, as it is shown in Figures 3B-C. Once the cyclic group to be tested is specified by the user, the software automatically tests each rotation operator of this group, and provides the best rotation axis and the resulting RMSD. We should specifically mention that in most of the practical cases we have assemblies with only two subunits. In this case, there is only one rotation operator that superposes the present subunits and the comparisons presented in Figure 3A and 3C will be equivalent to each other. In the examples we have encountered, the different results coming from the choice of different rotation operators are very close to each other, and in the case where the symmetry is perfect, any chosen method will provide exactly the same result.

3. Results and Discussion

3.1. Pseudo-Symmetrical C_n examples

We will first demonstrate our method on complete pseudo-symmetrical assemblies, for which we will determine the axis of symmetry and the RMSD measure. Pseudo-symmetrical assemblies are complexes that look symmetrical, however their sequences in different subunits are not the same. For the following example we have picked one pseudo-symmetrical assembly from

Order	RMSD (Å)	Axis
C_4	12.39	(0.986, 0.161, -0.050)
C_5	5.61	(0.991, 0.129, -0.036)
C_6	2.34	(0.994, 0.110, -0.030)
C_7	3.93	(0.995, 0.097, -0.027)
C_8	6.20	(0.996, 0.089, -0.025)

Table 1: RMSD symmetry measures and the symmetry axes computed for several symmetry orders of the 2GZA structure.

each of C_2, C_3, C_5, and C_8 cyclic groups that are available in PDB. The PDB codes of these assemblies are 1AOK, 2WNV, 5KXI, and 3IYG, correspondingly. Figure 4 shows the output of our method. The RMSD symmetry measures for these assemblies are 1.406 Å, 2.226 Å, 1.613 Å, and 2.736 Å, correspondingly. The determined symmetry axes are shown with polygons.

3.2. Reconstruction of assemblies with missing subunits

In the following example we will illustrate the possibility of finding the axis of symmetry of a partial assembly that does not pass through its COM. For this purpose we will consider the PDB structure 2GZA. The asymmetric subunit of this structure contains three chains with identical sequence and crystallographic information explains that this subunit should be replicated two times around the x-axis to obtain the biological assembly.

From the three chains in the PDB file, we computed the RMSD for cyclic symmetries of different order. Table 1 lists the obtained results. We can see that the asymmetric unit present in the PDB file is consistent with a C_6 symmetry (RMSD of 2.34 Å), but a C_7 symmetry (RMSD of 3.93 Å) could also be possible. We should also mention that the found axes of symmetry are rather different from the x-axis provided by the crystallographic information. For example, for the C_6 case, the two axes have about 6 degrees of difference. Using the computed axes, we can also reconstruct the C_6 and C_7 assemblies by a replication of the asymmetric unit for the C_6 case, and a replication of the asymmetric unit plus one more chain for the C_7 case. Figures 5B-C show the obtained assemblies. If we compute RMSDs for the reconstructed assemblies, we obtain the values of 2.74 Å for the C_6 reconstruction (Fig. 5B), 4.24 Å for the C_7 reconstruction (Fig. 5C), and 4.85 Å for the reconstruction from crystallographic information (Fig. 5A). The

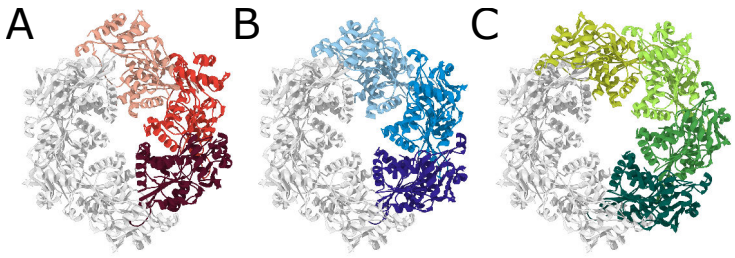

Figure 5: Cyclic reconstructions of the PDB structure 2GZA. The grey color corresponds to the asymmetric unit, which consists of three chains. A - In red we show the reconstruction of the assembly based on the crystallographic information. The corresponding RMSD measure is 4.85 Å. B - In blue we show the reconstruction made with the optimal C_6 axis. The corresponding RMSD measure is 2.74 Å. C - In green we show the reconstruction made with the optimal C_7 axis. The corresponding RMSD measure is 4.24 Å.

big difference between the symmetry measures obtained by reconstruction with and without the crystallographic information, and the fact that in a crystal this assembly is less symmetric than the C_7 reconstructed version, may suggest that this protein forms a C_7 assembly in solution and is forced to be in a C_6 conformation in a crystal.

3.3. Generation of perfectly symmetrical assemblies

A particularly interesting task in molecular modeling and crystallographic applications is to use an approximately symmetrical assembly as a starting model and generate a perfectly symmetrical structure from it. As a starting structure one can use an assembly from molecular dynamics simulations, a pseudo-symmetrical assembly, or the one with non-crystallographic symmetry, for example. Then, we proceed by computing the best C_n axis from the initial model. After, we choose one of the subunits as a 'master' subunit and replicate it around this axis to obtain the perfectly symmetrical assembly. Figure 6 illustrates this approach when using a pseudo-symmetrical C_3 assembly (PDB code 2IX2) as an input structure. This structure is composed of three chains with two different sequences. The RMSD measure of this structure is 6.20 Å. The symmetrized assembly is perfectly symmetrical and obviously has the RMSD measure of 0 Å.

3.4. Comparison with other methods

In order to demonstrate the efficiency of our approach, we compared it with two other published techniques. The first one was developed by Dryzun

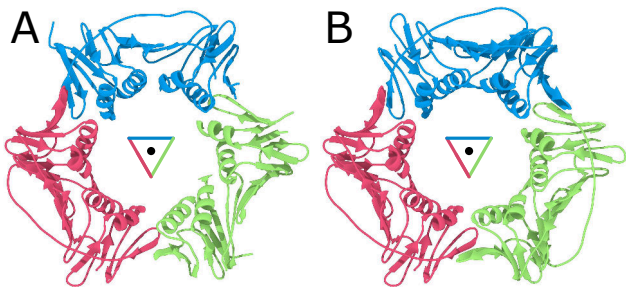

Figure 6: A - A pseudo-symmetrical C_3 assembly (PDB code 2IX2) with the axis of symmetry shown with the triangle. Its three chains are shown with three different colors and are slightly different from each other. B - The symmetrized version of this assembly. Here, we arbitrarily chose the red chain from the complex in A and replicated it to obtain the perfectly symmetrical assembly.

et al. [28], and we will refer to it as to CSM (Continuous Symmetry Measure). It considers all the atoms in the input assembly and finds the symmetry axis by alternatively refining the axis of rotation and the permutation between the atoms. Table 2 lists all the cyclic examples found in the CSM article [28]. We should note that Dryzun et al. [28] report either the symmetry measure or the computational time. The CSM symmetry measure can easily be converted to the RMSD symmetry measure by the following equation,

$$\mathrm{RMSD}^2 = \frac{\mathrm{CSM} \times R_g^2}{50}, \tag{28}$$

where R_g is the radius of gyration of the assembly. The second method is from Levy et al. [2], and will be called Levy. It exhaustively scans a finite set of axes of symmetry and chooses the best one. Unlike the previous technique, it has to be fed with lists of atoms organized in subunits. Therefore, to prepare the input, we used the same alignment procedure as we implemented in our method, and we used parameters suggested by the author.

Table 2 lists the execution time and the symmetry measure (RMSD value) for the three tested methods. It shows that our method scales with the size of the input assembly much better than the two other methods. Indeed, its runtime typically stays below one second, even for large assemblies.

On all the tested examples, our method is significantly faster than the one from Levy and it also produces a lower RMSD measure. In practice, we obtain the same RMSD when the actual symmetry axis is among the ones

PDB Code	Group	RMSD(AnAnaS)	RMSD(CSM)	RMSD(Levy)	AnAnaS Time[a]	CSM Time[b]	Levy Time[a]
1HPV	C_2	0.23 Å	-	0.23 Å	0.02 s	1.9 s	0.11 s
1LGN	C_5	0.20 Å	-	0.36 Å	0.15 s	34 s	1.02 s
1NN2[c]	C_4	0.00 Å	-	0.00 Å	0.19 s	77 s	0.77 s
2FKW	C_9	0.28 Å	-	0.81 Å	0.15 s	1175 s	3.9 s
2XE2	C_3	0.12 Å	0.23 Å	0.12 Å	0.11 s	-	0.42 s
3FV9	C_8	27.7 Å	19.8 Å	>7 Å	0.73 s	-	7.32 s
3FV9	C_4	0.48 Å	7.6 Å	0.60 Å	0.73 s	-	1.36 s
3KML	C_{17}	0.36 Å	0.45 Å	0.67 Å	1.7 s	-	74 s

[a] AnAnaS and Levy times were measured on a Windows laptop equipped with an Intel i7 @ 3.1 GHz.
[b] CSM times were taken from [28] with a different, a 7 year older, CPU. However, we believe that the order of magnitude of these timings is still correct.
[c] For this structure, the biological assembly was used.

Table 2: Comparative results between AnAnaS, CSM and Levy methods tested on cyclic examples collected from the CSM paper [28].

sampled by Levy's method. Comparison to CSM is a bit more difficult because this method considers more atoms (reference points) than we do, and also because we do not have the computed axes for the analysis. These additional atoms can explain small differences in the computed RMSD values. We should note that more freedom in choosing the correspondence between the atoms can significantly lower RMSD in poorly symmetrical assemblies. These two effects explain the small differences in the 2XE2 and 3KML examples, and also the difference in the 3FV9 example when measuring the C_8 symmetry. However, we believe that the iterative process of CSM was stuck in a local minimum when measuring the C_4 symmetry. Indeed, visual inspection reveals that the 3FV9 assembly has a D_4 symmetry that seems of a very high quality, thus it is not possible that the average deviation between the different dimers is more than 7Å, as reported by CSM. In this example, the dihedral symmetry makes the 4-fold axis much more difficult to detect by CSM, because several 2-fold axes are also present.

3.5. Computational Details

We implemented the method using the C++ programming language. The method is called AnAnaS, which stands for Analytical Analysis of Symmetries. It is available as a standalone executable and also as a module with graphical user interface for the SAMSON software platform. We can also provide the source code upon request.

The most time consuming part of the method is the multiple sequence alignment required to compare the relevant alpha carbons in different subunits, which typically takes time from a few milliseconds to a few seconds.

This sequence alignment can be seen as a potential weakness in the procedure as it is not analytical. However, for homomeric assemblies, which are the most common ones, the alignment is trivial since all the chains have the same sequence. The alignment also prevents from comparing unrelated parts of different chains. Finally, it significantly reduces the number of possible matches between atoms in different chains and makes the method robust against inconsistencies in the input data.

Then, the formulation of the optimization problem takes time linear with the number of matched atoms, typically a few milliseconds. Finally, solution of the constrained quadratic optimization problems 26 takes only constant time and the solver of the trust-region subproblem converges to machine precision in 3-10 iterations, which takes a few microseconds.

We should also add that our method successfully perceives cyclic symmetries within higher-order symmetrical assemblies, such as dihedral and cubic. This perception is based on a robust determination of permutations between the assembly subunits corresponding to each rotation operator within the symmetry group. All the relevant details including the discrete optimization approach for the identification of the permutations are described in the second part of this work, which is specifically devoted to high-order symmetries with multiple symmetry axes [26].

4. Conclusions

This work presents an efficient computational approach to assess the quality of cyclic C_n symmetry in macromolecular assemblies. We express the quality through the symmetry measure using a Euclidian 3D distance. We showed that the problem of finding the best symmetry axis can be formulated as a constrained quadratic optimization problem and provided an efficient solution to it. More precisely, using the quaternion arithmetic, we expressed the rotation operators through quadratic forms with constraints. This allowed us to find the unique solution using efficient methods developed for the trust-region sub-problem. We have demonstrated the efficiency of the method on several examples including partial assemblies and pseudo symmetries. We have also compared the presented method with two other published techniques and showed that out method is significantly faster on all the tested examples.

In the second part of this work, we will tackle a more challenging case of dihedral and cubic symmetry groups, and provide a general analysis of

all the symmetrical assemblies found in the PDB. The method is available at https://team.inria.fr/nano-d/software/ananas/. The SAMSON module is available at http://samson-connect.net/.

5. Aknowledgements

The authors thank Alexandre Hoffmann from the Nano-D team of Inria Grenoble for his help with the solution of the trust-region subproblem. The authors also thank Andrey Kazennov from MIPT Moscow for his support at the initial stage of the project.

Funding

This work was supported by L'Agence Nationale de la Recherche (grant number ANR-15-CE11-0029-03), and by the 5TOP100 program of the Ministry for Education and Science of the Russian Federation.

References

[1] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. Bhat, H. Weissig, I. N. Shindyalov, P. E. Bourne, The Protein Data Bank, Nucleic Acids Res 28 (2000) 235–242.

[2] E. D. Levy, J. B. Pereira-Leal, C. Chothia, S. A. Teichmann, 3D complex: a structural classification of protein complexes, PLoS Comput Biol 2 (2006) e155.

[3] E. D. Levy, E. B. Erba, C. V. Robinson, S. A. Teichmann, Assembly reflects evolution of protein complexes, Nature 453 (2008) 1262–1265.

[4] D. W. Ritchie, S. Grudinin, Spherical polar fourier assembly of protein complexes with arbitrary point group symmetry, J Appl Crystallogr 49 (2016) 158–167.

[5] D. Lukatsky, B. Shakhnovich, J. Mintseris, E. Shakhnovich, Structural similarity enhances interaction propensity of proteins, J Mol Biol 365 (2007) 1596–1606.

[6] I. André, C. E. Strauss, D. B. Kaplan, P. Bradley, D. Baker, Emergence of symmetry in homooligomeric biological assemblies, Proc Natl Acad Sci USA 105 (2008) 16148–16152.

[7] G. E. Schulz, The dominance of symmetry in the evolution of homo-oligomeric proteins, J Mol Biol 395 (2010) 834–843.

[8] E. D. Levy, S. Teichmann, Structural, evolutionary, and assembly principles of protein oligomerization, Prog Mol Biol Transl 117 (2013) 25–51.

[9] S. E. Ahnert, J. A. Marsh, H. Hernández, C. V. Robinson, S. A. Teichmann, Principles of assembly reveal a periodic table of protein complexes, Science 350 (2015) aaa2245.

[10] H. Garcia-Seisdedos, C. Empereur-Mot, N. Elad, E. D. Levy, Proteins evolve on the edge of supramolecular self-assembly, Nature 548 (2017) 244.

[11] T. L. Blundell, N. Srinivasan, Symmetry, stability, and dynamics of multidomain and multicomponent protein systems, Proc Natl Acad Sci USA 93 (1996) 14243–14248.

[12] D. S. Goodsell, A. J. Olson, Structural symmetry and protein function, Annu Rev Bioph Biom 29 (2000) 105–153.

[13] K. B. Murray, W. R. Taylor, J. M. Thornton, Toward the detection and validation of repeats in protein structure, Proteins 57 (2004) 365–380.

[14] D. Myers-Turnbull, S. E. Bliven, P. W. Rose, Z. K. Aziz, P. Youkharibache, P. E. Bourne, A. Prlić, Systematic detection of internal symmetry in proteins using CE-Symm, J Mol Biol 426 (2014) 2255–2268.

[15] H. Chen, Y. Huang, Y. Xiao, A simple method of identifying symmetric substructures of proteins, Comput Biol Chem 33 (2009) 100–107.

[16] E. S. Shih, M.-J. Hwang, Alternative alignments from comparison of protein structures, Proteins 56 (2004) 519–527.

[17] M. Pinsky, C. Dryzun, D. Casanova, P. Alemany, D. Avnir, Analytical methods for calculating continuous symmetry measures and the chirality measure, J Comput Chem 29 (2008) 2712–2721.

[18] M. Pinsky, A. Zait, M. Bonjack, D. Avnir, Continuous symmetry analyses: Cnv and dn measures of molecules, complexes, and proteins, J Comput Chem 34 (2013) 2–9.

[19] C. Dryzun, Continuous symmetry measures for complex symmetry group, J Comput Chem 35 (2014) 748–755.

[20] M. Petitjean, On the root mean square quantitative chirality and quantitative symmetry measures, J Math Phys 40 (1999) 4587–4595.

[21] B. K. Horn, Closed-form solution of absolute orientation using unit quaternions, J Opt Soc Am A 4 (1987) 629–642.

[22] R. Diamond, A note on the rotational superposition problem, Acta Crystallogr A 44 (1988) 211–216.

[23] S. K. Kearsley, On the orthogonal transformation used for structural comparisons, Acta Crystallogr A 45 (1989) 208–210.

[24] P. Popov, S. Grudinin, Rapid determination of rmsds corresponding to macromolecular rigid body motions, J Comput Chem 35 (2014) 950–956.

[25] E. Neveu, P. Popov, A. Hoffmann, A. Migliosi, X. Besseron, G. Danoy, P. Bouvry, S. Grudinin, RapidRMSD : Rapid determination of RMSDs corresponding to motions of flexible molecules, 2018. In Press, bty160, https://doi.org/10.1093/bioinformatics/bty160.

[26] G. Pagès, S. Grudinin, Analytical symmetry detection in protein assemblies. II. Dihedral and Cubic symmetries, Unpublished.

[27] D. C. Sorensen, Newton's method with a model trust region modification, SIAM J Numer Anal 19 (1982) 409–426.

[28] C. Dryzun, A. Zait, D. Avnir, Quantitative symmetry and chirality—a fast computational algorithm for large structures: Proteins, macromolecules, nanotubes, and unit cells, J Comput Chem 32 (2011) 2526–2538.